Informatik
Hauptcampus

HOCHSCHULE TRIER

Prozedurale Generierung von Baum-Strukturen innerhalb der Unreal Engine 4

Procedural generation of tree-structures in the Unreal Engine 4

David Liebemann

Bachelor-Abschlussarbeit

Betreuer:
Prof. Dr. Christof Rezk-Salama

Trier, 26.02.2017

Kurzfassung

Inhaltsverzeichnis

Abbildungsverzeichnis

1

Einleitung und Problemstellung

1.1 Prozedurale Generierung

Insbesondere Vegetation - warum, was sind die Schwierigkeiten, wenn man es nicht prozedural generiert.

1.2 Bisherige Arbeiten

1.3 Unreal Engine 4

Allgemeine Erklärung - was ist die Unreal Engine 4, was stellt sie zur Verfügung, wie entwickelt man dafür? Insbesondere Eingabe von Parametern über den Editor –> wichtig für Implementierung.

1.4 Ansatz

Verwendete Ansätze: L-Systeme mit Turtle-Graphik und Space Colonization Algorithmus. Kleine Einleitung zu beiden Ansätzen, bisherige Verwendung.

2

Lindenmayer-Systeme

Vorstellung der verwendeten Konzepte.

2.1 Kontextfreie Grammatik

Bei L-Systemen handelt es sich um, auf Zeichenketten basierende, Ersetzungssysteme. Es werden komplexe Objekte beschrieben, indem Teile der Zeichenkette durch andere Zeichen oder Zeichenketten ersetzt werden. Die Beschreibung dieser Ersetzungen findet mittels festgelegter Produktionsregeln statt. [PL04, S.2]

Eine formale Definition eines auf Zeichenketten arbeitenden Ersetzungssystems wird durch kontextfreie Grammatiken gegeben:

Kontextfreie Grammatik: 2.1 *Eine kontextfreie Grammatik G ist ein Tupel G* $= (V, N, P, \omega)$ *bestehend aus:*

1. V – *einer nichtleeren, endlichen Menge von Buchstaben (Alphabet).*
2. N – *einer endlichen Menge von Variablen.*
3. P – *einer endlichen Menge von Produktionsregeln in der Form* $P : A \to \alpha$ *mit* $A \in N$ *und* $\alpha \in (V \cup N)^*$.
4. $\omega \in N$ – *dem Axiom, Startsymbol der Grammatik.*

[Sch14, S.343]

Die Menge V^* ist die Menge aller Wörter über V – die Menge aller Wörter, die aus dem Alphabet V gebildet werden können. [Sch14, S.70]

Eine Grammatik wird als kontextfrei bezeichnet, wenn beispielsweise die Produktionsregel $A \to \alpha$ angewendet werden kann, ohne die A umgebenden Buchstaben – seinen Kontext – beachten zu müssen. [Sch14, S.343]

Eine Grammatik wird als deterministisch bezeichnet, wenn es genau eine Produktionsregel $r \in P$ für jede Variable $A \in V$ gibt, sodass $r : A \to \alpha$, $\alpha \in (V \cup N)^*$. Das bedeutet, dass die Ersetzung einer Variable eindeutig durch eine einzige Regel beschrieben wird. [STN16, S.75]

Die Anwendung der Produktionsregeln findet meist sequentiell statt – die Zeichenkette wird von links nach rechts durchlaufen und Ersetzungen werden direkt auf die untersuchte Zeichenkette angewendet. [STN16, S.75]

2.2 D0L-Systeme

Diese Arbeit beschränkt sich auf die Behandlung deterministischer, kontextfreier L-Systeme, auch D0L-Systeme genannt. Diese besitzen die Eigenschaften einer deterministischen und kontextfreien Grammatik, Produktionsregeln werden jedoch parallel und gleichzeitig auf alle Buchstaben des untersuchten Wortes angewendet. Dieses Vorgehen soll die Zellteilung in mehrzelligen Organismen simulieren und ist somit an biologische Vorgänge angelehnt. [PL04, S. 3]

Ein D0L-System kann wie folgt definiert werden:

D0L-System: 2.1 *Ein D0L-System ist ein Tupel* $G = (V, P, \omega)$*, bestehend aus*

1. V *– einem nichtleeren, endlichen Alphabet.*
2. P *– einer endlichen Menge von Produktionsregeln in der Form* $P : a \to b$ *mit* $a \in V$ *und* $b \in V^*$*.* a *wird als Vorgänger,* b *als Nachfolger bezeichnet. Ist für einen Buchstaben* $x \in V$ *keine explizite Produktionsregel angegeben, wird die Identitätsproduktion* $P : x \to x$ *angenommen – der Buchstabe wird durch sich selbst ersetzt.*
3. $\omega \in V^+$ *– dem Axiom, Startsymbol der Grammatik.*

Die Menge V^+ *ist die Menge aller nichtleeren Wörter über* V*. [PL04, S.4]*

Die Ableitung eines Wortes entspricht der Ersetzung aller Buchstaben anhand der Produktionsregeln. Ein Wort kann mehrmals abgeleitet werden.

Ableitung: 2.1 *Gegeben sei ein Wort* $w = a_1 ... a_m$ *mit* $w \in V^*$ *und* $a_i \in V^*$*. Das Wort* $v = b_1 ... b_m$ *mit* $v \in V^*$ *und* $b_i \in V^*$ *ist die Ableitung von* w *wenn für alle* $i = 1...m$ *eine Produktionsregel* $a_i \to b_i$ *existiert. Die Ableitung wird als* $w \Rightarrow v$ *notiert.*
Das Wort w_n *ist die* n*-te Ableitung des Wortes* w_0 *wenn eine Folge von Wörtern* $w_0, w_1, ..., w_n$ *mit Ableitungen* $w_0 \Rightarrow w_1 \Rightarrow ... \Rightarrow w_n$ *existiert. [PL04, S.4]*

Anabaena

Beispiel: Das Wachstum der Blaualgen-Gattung „Anabaena“ kann durch ein L-System simuliert werden. Die Buchstaben a und b beschreiben die Größe und Teilungsbereitschaft einer Algenzelle, während die Indizes l und r die Polarität einer Zelle darstellen. Es gelten folgende Produktionsregeln:

$$\begin{aligned} p_1 &: a_r \to a_l b_r \\ p_2 &: a_l \to b_l a_r \\ p_3 &: b_r \to a_r \\ p_4 &: b_l \to a_l \end{aligned} \tag{2.1}$$

Die Entwicklung einer Anfangszelle a_r (Axiom $\omega : a_r$) läuft daraufhin wie in Abbildung 2.1 dargestellt ab.

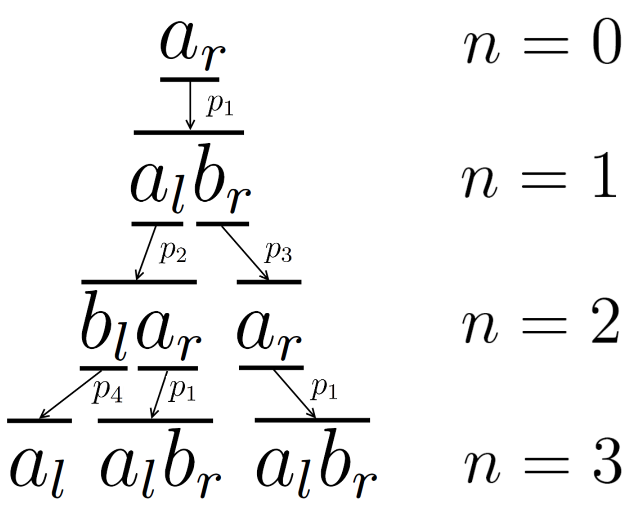


Abb. 2.1: n-fache Ableitung der Anfangszelle a_r anhand der Produktionsregeln $p_1 ... p_4$ aus Gleichung 2.1. Eigene Abbildung anhand [PL04, S.4]

2.2.1 Parametrische L-Systeme

Parametrische L-Systeme stellen eine Erweiterung der D0L-Systeme dar. Die Buchstaben eines verwendeten Alphabets V werden um zugeordnete Parameter aus der Menge der reelen Zahlen ergänzt. Ein solches parametrisches Wort $V \times \Re^*$ besteht aus einem Zeichen $A \in V$ und Parametern $a_1, ..., a_n \in \Re$ und wird als $A(a_1, ..., a_n)$ dargestellt. Ein parametrisches Wort ohne Parameter mit dem Zeichen $A \in V$ wird schlicht als A dargestellt. [PL04, S.41]

Während in der Definition von Modulen numerische Konstanten verwendet werden, werden bei der Angabe eines L-Systems Variablen verwendet. Ist Σ eine Menge von Variablen, dann ist $E(\Sigma)$ ein arithmetischer Ausdruck, in dem Variablen, Konstanten und arithmetische Operatoren auf eine zulässige Weise kombiniert werden. [PL04, S.41]

Parametrisches L-System: 2.2.1 *Ein Parametrisches L-System ist ein Tupel G* $= (V, \Sigma, P, \omega)$, *bestehend aus*

1. V – *einem nichtleeren, endlichen Alphabet.*
2. Σ – *einer Menge der Variablen.*
3. P – *einer endlichen Menge von Produktionsregeln* $P : (V \times \Sigma^*) \rightarrow (V \times E(\Sigma))^*$
4. $\omega \in M^+$ *mit* $M = (V \times \Re^*)$ – *einem Axiom in Form eines nichtleeren, parametrischen Wortes.*

[PL04, S.41]

Eine Produktion kann auf ein parametrisches Wort angewendet werden wenn das Zeichen, das dem Wort vorausgeht, und die Anzahl der Parameter mit dem Zeichen und der Anzahl der Parameter im Vorgänger der Produktionsregel übereinstimmen. [PL04, S.42]

Beispiel: Gegeben sei folgendes, parametrisches L-System:

$$\begin{aligned} \omega\ &: A(1,1) \\ p_1 &: A(x,y) \rightarrow A(x+1, y*2)\ B(y) \\ p_2 &: B(x) \quad\ \rightarrow B(x+1)\ C \end{aligned} \tag{2.2}$$

Das Alphabet V und die Menge der Variablen Σ gehen implizit aus der Angabe der Produktionsregeln hervor. Die Entwicklung des L-Systems läuft anhand Abbildung 2.2 ab.

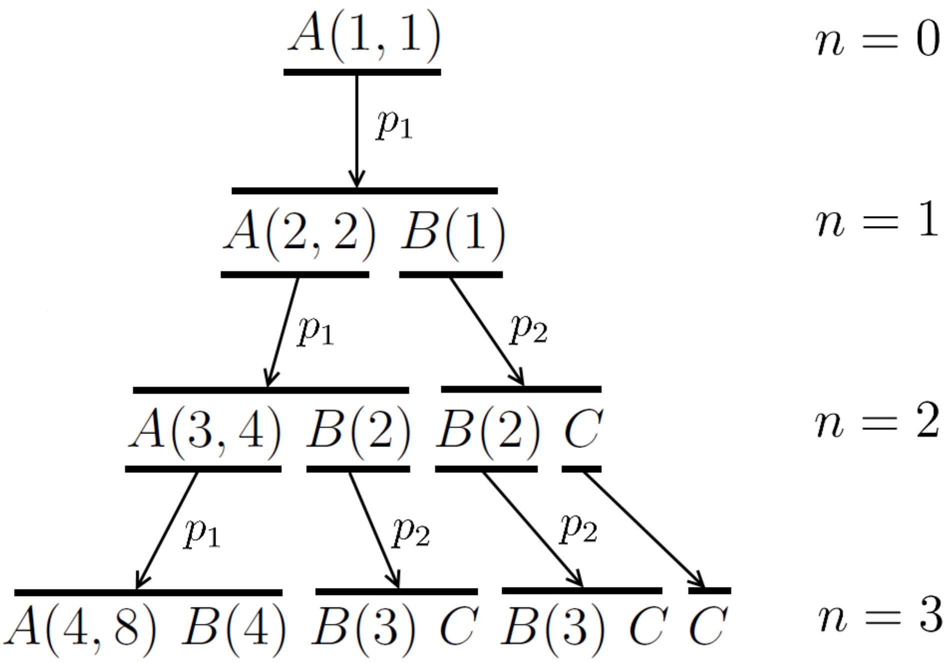


Abb. 2.2: n-fache Ableitung des Axioms $A(1,1)$ anhand der Produktionsregeln aus Gleichung 2.2. Die Ersetzung des Zeichens C erfolgt anhand der impliziten Identitätsproduktion. Eigene Abbildung.

2.3 Grafische Interpretation von L-Systemen

Um die Ergebnisse von L-Systemen in Form von dreidimensionalen Objekten zu visualisieren, muss eine grafische Interpretation der resultierenden Zeichenketten festgelegt werden. Im Folgenden wird die verwendete Visualisierungsmethode, Turtle-Interpretation, vorgestellt.

2.3.1 Turtle-Interpretation

Die Turtle-Interpretation im zweidimensionalen Raum entspricht der Vorstellung einer Turtle[1] auf einem Blatt Papier. Die Turtle besitzt eine Position $\overrightarrow{P}$ welche

[1] „Turtle“ – englisch für „Schildkröte“.

durch die kartesischen Koordinaten (p_1, p_2) beschrieben wird sowie einen Einheitsvektor $\overrightarrow{H}$ mit kartesischen Koordinaten (h_1, h_2). $\overrightarrow{P}$ beschreibt die Position der Turtle auf der Ebene und $\overrightarrow{H}$ entspricht der Blickrichtung (Heading) der Turtle. Der Zustand $(\overrightarrow{P}, \overrightarrow{H})$ einer Turtle wird somit vollständig durch die Position und Blickrichtung definiert. [GSJ04, S.2] Auf der Turtle können drei Aktionen durchgeführt werden, die durch die folgenden Symbole dargestellt werden:

$F(a)$ Die Turtle bewegt sich um $a > 0$ in die Richtung der aktuellen Blickrichtung. Die neue Position ist $\overrightarrow{P'} = (p'_1, p'_2)$ mit:

$$\begin{aligned} p'_1 &= p_1 + h_1 * a \\ p'_2 &= p_2 + h_2 * a \end{aligned} \tag{2.3}$$

Zwischen der alten Position $\overrightarrow{P}$ und der neuen Position $\overrightarrow{P'}$ wird eine Linie gezeichnet.

$+(a)$ Die Turtle dreht sich um den Winkel a nach links. Die neue Blickrichtung ist $\overrightarrow{H'} = (h'_1, h'_2)$ mit:

$$(h'_1, h'_2) = (h_1, h_2) * \begin{pmatrix} cos(a) & sin(a) \\ -sin(a) & cos(a) \end{pmatrix} \tag{2.4}$$

$-(a)$ Die Turtle dreht sich um den Winkel a nach rechts. Die neue Blickrichtung ist $\overrightarrow{H'} = (h'_1, h'_2)$ mit:

$$(h'_1, h'_2) = (h_1, h_2) * \begin{pmatrix} cos(-a) & sin(-a) \\ -sin(-a) & cos(-a) \end{pmatrix} \tag{2.5}$$

[GSJ04, S.4,46] [PL04, S.7] Die Symbole „+“ und „-“ werden sowohl im Alphabet eines L-Systems als auch bei arithmetischen Operationen in Parameterangaben verwendet, ihre Bedeutung ist abhängig vom Kontext, in welchem sie angewendet werden. [PL04, S.46]

Die grafische Turtle-Interpretation einer Zeichenkette, die durch ein L-System zurückgegeben wird, sind somit die Linien, die auf Grundlage der definierten Symbole gezeichnet werden.

Beispiel: Mithilfe von L-Systemen und einer Turtle-Interpretation können Fraktale, in diesem Beispiel sogenannte Koch-Kurven visualisiert werden. Diese Kurven bestehen aus einem Initiator – einer einfachen, zweidimensionalen Form – und einem Generator, der einem offenen Polygonzug entspricht. In jedem Ableitungsschritt, angefangen bei dem Initiator, wird jede gerade Linie durch den Generator ersetzt. [Man83, S.39]

Dieses Verhalten kann nun auf ein L-System abgebildet werden: Der Initiator entspricht dem Axiom und der Generator einer Produktionsregel des L-Systems. Der in Abbildung 2.3 dargestellte Initiator und Generator entsprechen der Turtle-Interpretation des folgenden L-Systems:

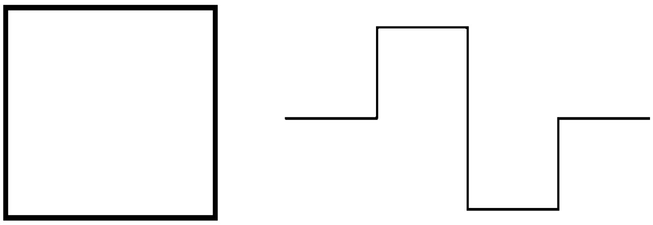
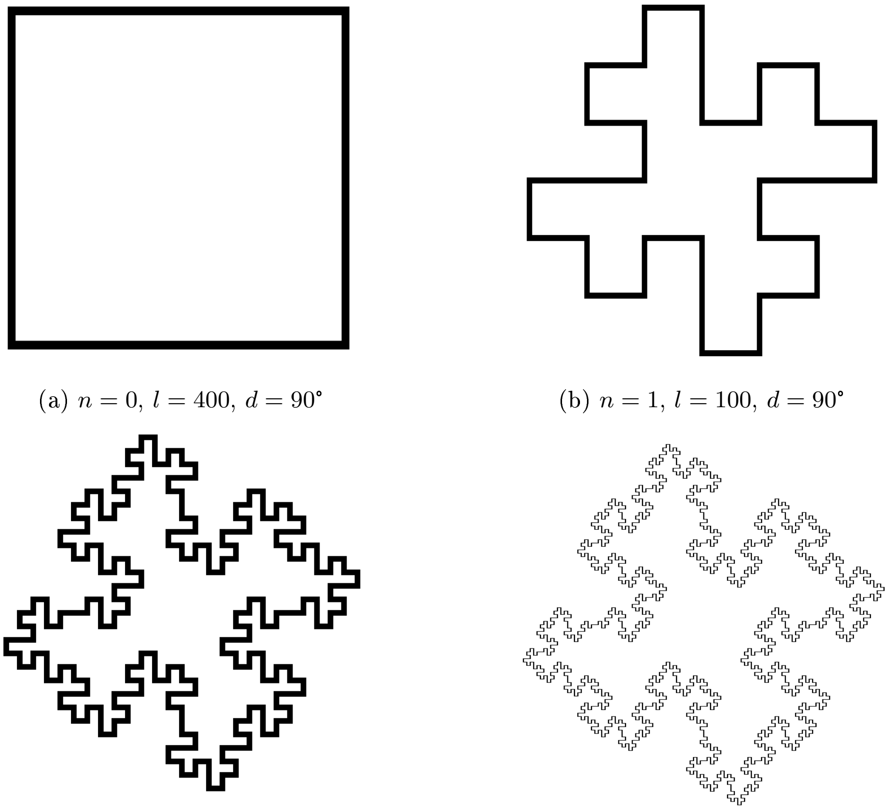


Abb. 2.3: Links: Initiator der Koch-Kurve in Form eines einfachen Quadrats. Rechts: Generator der Koch-Kurve in Form eines offenen Polygonzugs. Eigene Abbildung.

$$\begin{aligned} \omega\ &: F - F - F - F \\ p_1 &: F \rightarrow F + F - F - FF + F + F - F \end{aligned} \tag{2.6}$$

Für eine bessere Übersicht wurde die Angabe der Parameter weggelassen. Die Turtle interpretiert F als $F(l)$, $-$ als $-(d)$ und $+$ als $+(d)$ mit festgelegter Strichlänge l und Drehwinkel d. Die Entwicklung des L-Systems läuft anhand Abbildung 2.4 ab.



(a) $n = 0$, $l = 400$, $d = 90°$

(b) $n = 1$, $l = 100$, $d = 90°$

(c) $n = 2$, $l = 25$, $d = 90°$

(d) $n = 3$, $l = 6.25$, $d = 90°$

Abb. 2.4: n-fache Ableitung des Axioms ω anhand der Produktionsregel p_1 aus Gleichung 2.6, visualisiert mithilfe der Turtle-Interpretation. Eigene Abbildung.

2.3.2 Verzweigte L-Systeme

asd

2.3.3 Erweiterung in den dreidimensionalen Raum

2.4 L-Systeme für Baumstrukturen

asd

3

Space-Colonization Algorithmus

3.1 Ursprung

3.2 Erweiterung in dreidimensionalen Raum

3.3 Einfluss der Parameter

4

Implementierung

4.1 Baumstruktur

4.2 L-Systeme

4.2.1 L-System-Plant

4.2.2 Turtle-Graphic-Interpreter

4.2.3 Performanz

4.3 Space-Colonization-Algorithmus

4.3.1 Colonization Space

4.3.2 Space-Colonization-Plant

4.4 Mesh-Generierung

5

Ergebnisse und Vergleich

5.1 Visuell

5.2 Performanz

6

Zusammenfassung und Ausblick

Zusammenfassung

6.1 Erweiterungen

6.1.1 Texturen

6.1.2 Blätter

6.1.3 Generierung zur Laufzeit

6.1.4 Verteilung

Literaturverzeichnis

GSJ04. Goldman, Ron, Scott Schaefer und Tao Ju: *Turtle Geometry in Computer Graphics and Computer Aided Design*, 2004. `http://www.cs.wustl.edu/~taoju/research/TurtlesforCADRevised.pdf`.

Man83. Mandelbrot, Benoit B.: *The Fractal Geometry of Nature.* W. H. Freeman and Company, 1983.

PL04. Prusinkiewicz, Przemyslaw und Aristid Lindenmayer: *The Algorithmic Beauty of Plants.* Przemyslaw Prusinkiewicz, eBook Auflage, 2004. `http://algorithmicbotany.org/papers/abop/abop.pdf`.

Sch14. Schmitz, Prof. Dr. Heinz: *Theoretische Informatik - Vorlessungsskript*, 2014.

STN16. Shaker, Noor, Julian Togelius und Mark J. Nelson: *Procedural Content Generation in Games.* Springer International Publishing Switzerland 2016, 2016.

A

Glossar

B

Erklärung der Kandidatin / des Kandidaten

☐ Die Arbeit habe ich selbstständig verfasst und keine anderen als die angegebenen Quellen- und Hilfsmittel verwendet.

☐ Die Arbeit wurde als Gruppenarbeit angefertigt. Meine eigene Leistung ist ...

Diesen Teil habe ich selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet.

Namen der Mitverfasser: ...

Datum	Unterschrift der Kandidatin / des Kandidaten